International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-2 Issue-2, December 2012

Automation of Screen-Shot Analysis for Anti-Virus Toaster Windows

Sachin Jadhav, Shrikant Ganmukhe, Sanket Badwe, Bhushan Bhavsar

Abstract*—There are many antivirus products available in market. They provide different type of security levels to the user's data. For their own improvement they need to compare their product with their competitors to know the difference of security levels detected for the same type malware. To do such comparisons, the companies need to analyse the actions taken by the antivirus with toaster window displayed on desktop and hence they need to compare a large number of screen-shots of those actions. This project is used for automation of all these process to provide effective and better way of screen shot analysis by extracting text from them. Hence, the purpose of this project is to analyse and classify the actions taken by an antivirus for particular malware with the help of screen shots of those actions. It reduces the manual efforts and provides an automated way recognizing the activities done by an antivirus.

I. INTRODUCTION

The Antivirus products display a toaster (pop-up window) whenever a threat is detected on the system. In most cases, the threat is automatically removed by the antivirus and the pop-up window indicates the details of the threat e.g. the name of the file which was infected, the threat-type encounter and the action antivirus has performed (cleaned or deleted). The desktop screenshots for all these activities are captured as "PNG" images. The requirement is to design a library which will take the screen-shot "PNG" image feeds and determine if the antivirus detected threat, threat type etc. by extracting the text from the images.

The manual analysis of the screenshots requires about 8 to 10 man hours. By automating the screenshot analysis using the image processing the manual analysis can be avoided which can increase the reliability of the results and also reduce the required time.Screenshot contains different types icon, different size windows, which contain the text with different fonts, font size, colour etc. So proposed system has to detect and extract text from images, analyse and classify that extracted text into different antivirus level.

Manuscript published on 30 December 2012.
***** Correspondence Author (s)
Mr. Sachin Jadhav, Professor, Department of computer engineering , PimpriChinchwad College of Engineering, Pune University, Pune, Maharashtra, India.
Mr. Shrikant Ganmukhe, Bachelor of computer engineering, PimpriChinchwad College of Engineering, Pune University, Pune, Maharashtra, India.
Mr. Sanket Badwe, Bachelor of computer engineering, PimpriChinchwad College of Engineering, Pune University, Pune, Maharashtra, India.
Mr. Bhushan Bhavsar, Bachelor of computer engineering, PimpriChinchwad College of Engineering, Pune University, Pune, Maharashtra, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

II. RELATED WORK

Several approaches for text detection in images and videos have been proposed up till now. Based on the methods being used to localize text regions, these approaches can be categorized into two main classes:
a) Connected component based method.
b) Texture based method.

a) Connected component based method:

This class of approach employs connected component analysis, which consists of analysing the geometrical arrangement of edges or homogeneous colour and grey-scale components that belong to characters. For example, *Cai et al.* have presented a text detection approach which is based on character features like edge strength, edge density and horizontal distribution. First, a colour edge detection algorithm is applied in YUV colour space and filter out non-text edges using a low threshold. Then, a local Thresholding technique is employed in order to keep low-contrast text and simplify the background. Finally, projection profiles are analysed to localize text regions.

b) Texture based methods:

The second class of approach regards texts as regions with distinct textural properties, such as character components that contrast the background and at the same time exhibit a periodic horizontal intensity variation, due to the horizontal alignment of characters. Methods of texture analysis like Gabor filtering and spatial variance are used to automatically locate text regions. Such approaches do not perform well with different character font sizes, and furthermore, they are computationally intensive

In our project, it is clear that the images used contain text of varying fonts and size. Since, the texture based method fails when the text is of different size and fonts, we are going to use connected component based method.

Hence, in this method we are going to convert colour image into grey scale format in order to locate the exact area where text is located regardless of the size and fonts used.

III. PROPOSED SYSTEM

Hence, the system we are going to develop is a testing tool doing all automated work for the toaster window analysis. The system will be having the screen shot of antivirus actions and it will be generating a well formed report of those actions as an output.

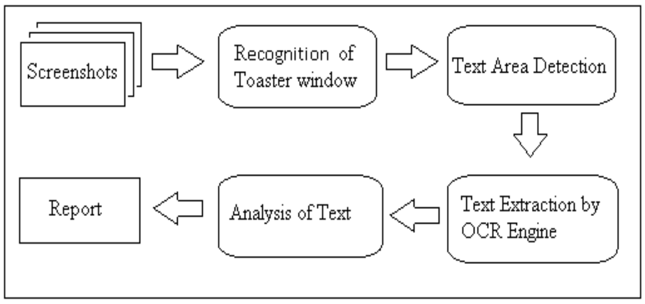

Figure 1: System Flow

- ***Screenshots:***

These are live desktop image taken in continuous feed and are input to system.

- ***Text Area Recognition:***

The system will be calculating the difference between two continuous images and finding the text prone area for reasonable difference. Then the area will be passed as an input to the text detection system for the further processing.

- ***Text Detection:***

This is the area where image processing will be carried out totally. The text will be detected with help of various types of supportive methods like edge detection, text region detection, text region segmentation etc.

- ***OCR Engine:***

The system will be using OCR engines for extracting the text. For this, the functionality of keyword filtering and matching it with pre-defined keywords will be carried out.

- ***Classification of Retrieved Text:***

There are various antivirus levels like Intrusion prevention, firewall, file reputation, behaviour and signature. The extracted text will be classified according to these types.

IV. PROPOSED ALGORITHM

Algorithm for Text extraction for antivirus toaster windows

Step 1.Selction of screenshot images

Take continues feed of screenshot images in .PNG format. Calculate difference between two consecutive images, if this value is greater than *delta*
then select that image for processing. Here *delta* is predefined value which indicate that there is change in next image.

Step 2.Detect Toaster window.

By calculating change in pixel value of two consecutive image it can detect coordinate of toaster window and pass only that toaster window for processing.

Step3.Text Detection by Connected Component based method.

3.1) Image processing.

Convert RGB image into YUV format.our system only uses the luminance data (Y channel of YUV) during further processing. After that, luminance value Thresholding is applied to spread luminance values throughout the image and increase the contrast between the possibly interesting regions and the rest of the image.

3.2) Edge Detection by Contrast Segmentation

We employ a simple method for converting the gray-level image into an edge image. This method is based on the fact that the character contours have high contrast to their local neighbours. As a result, all character pixels as well as some non-character pixels which also show high local colour contrast are registered in the edge image. In this image, the value of each pixel of the original image is replaced by the largest difference between itself and its neighbours (in horizontal, vertical and diagonal direction). Despite its simplicity, this procedures highly effective. Finally, the contrast between edges will be increased by means of a convolution with an appropriate mask.

```
Algorithm 3.1
EdgeImageDetection
// CreateedgeImage as ouput image in X×Y Format
//Input given is a grayImage generated in step 3.1
x = 0, y = 0,left = 0, upper= 0, rightUpper = 0
forallpixel(x,y)€ grayImagedo
        if(0<x<X-1)and(0<y<Y) then
        left = | pixel(x,y)- pixel(x-1,y) |
        upper = | pixel(x,y)- pixel(x,y-1) |
rightUpper = | pixel(x,y)- pixel(x+1,y-1) |
        edgeImage(x,y)= max(left,upper,rightUpper)
        else
        edgeImage(x,y)= 0
        end if
end for
edgeImage = sharpen(edgeImage)
rerutn(edgeImage)
```

Figure1: Algorithm for edge image detection

3.3) Detection of Text Region

The horizontal projection profile of the edge image is analysed in order to locate potential text areas. Since text regions show high contrast values, it is expected that they produce high peaks in horizontal projection. First, the histogram H is computed, whereel$_y$ is the number of pixels in line y of the edge image exceeding a given value. In subsequent processing, the local maxima are calculated by the histogram determined above. Two thresholds are employed to find the local maxima. A line of the images accepted as a text line candidate if either it contains a sufficient number of sharp edges or the difference between the edge pixels in one line to its previous line is bigger than a threshold. Both thresholds are defined empirically and are fixed. In this way, a text region is isolated which may contain several texts aligned horizontally.

```
Algorithm 3.2
textRegion[ ] detctTextRegion(Image edgeImage)
//Input given is edgeImage generated in algorithm 3.1
//textRegionis a data structure with 4 fields: x0,y0,x1,y1
//Determine Y-coordinates used by algorithm 3.3
//Determine X-coordinates used by algorithm 3.4
Integer H[ ] = calculateLineHistogram(edgeImage)
textRegion TC[ ] = determineYcoordinate(H)
TC= determineXcoordinate(edgeImage,TC)
return(TC)
```

Figure2: Algorithm for detection for text region

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

3.4) Enhancement & Segmentation of Text Region.

First, geometric properties of the text characters like thepossible height, width, width to height ratio are used to discard those regions whose geometric features do not fall into the predefined ranges of values. All remaining text candidates undergo another treatment in order to generate the so-called text image where detected text appears on a simplified background. The binary edge image is generated from the edge image, erasing all pixels outside the predefined textboxes and then binarizing it. This is followed by the process of gap filling. If one white pixel on the binary edge image is surrounded by two black pixels in horizontal, verticalor diagonal direction, then it is also filled with black. The gap image is used as a reference image to refine the localization of the detected text candidates. Text segmentation is the next step to take place. It starts with extraction of text candidates from the gray image. Then, the segmentation process concludes with a procedure which enhances text tobackground contrast on the text image.

```
Algorithm 3.3
textRegion[ ] determineYcoordinate(Integer H[ ])
// H is the line histogram
textRegionrect
textRegionTC[ ]
y = 1, j = 0
insideTextArea = FALSE
forel(y)€ H do
if ((el(y)>MinEdges)OR(el(y)- el(y-1)>MinLineDiff)) then
ifNOTinsideTextAreathen
        rect.y0 = y
        insideTextArea = TRUE
end if
elseifinsideTextAreathen
        rect.y1 = y-1
        if ((rect.y1 - rect.y0 >MinLines) then
                TC[j] = rect
                j = j+1
        end if
        insideTextArea = FALSE
end if
end for
return(TC)
```

Figure3: Algorithm for determining Y coordinates of text regions

Step 4.Pass the Image to OCR Engine.

The binary image is passed to OCR Engine which extracts text from image and correct errors in text.OCR engine perform different operation on images and gives digital text output.

Step 5.Classify the text in following antivirus level

i) IPS (Intrusion prevention System)
ii) Firewall
iii) File Reputation
iv) Behaviour
v) Signature

All extracted text is compared with some predefined keywords to classify the action taken by antivirus in different level.

```
Algorithm 3.4
textRegion[ ] determineXcoordinate(Image edgeImage, textRegion TC[ ])
left = maxInt,
right = -1
fortextCandidate_i€ TCdo
        forallpixel(x,y) € textCandidate_ido
                if(edgeImage(x,y) ≠ 0)then
                        if (left > x) then
                        left = x
                end if
                if (right < x) then
                        right = x
                end if
        end if
end for
textCandidate_i.x0= left
textCandidate_i.x1 =right
end for
return(TC)
```

Step 6. Generate result/report.

Report is generated in Excel spread sheet in following format.

Antivirus Name:

Sr. no	**Test Name**	**Image Name**	**Virus Info**	**Security Level**	**Action taken**
1.					

V. FEATURES

This project has following features which in turn describes the scope:

- It provides effective and automated way for screen-shot analysis.
- Helpful in comparing the antivirus products with any number of competitors.
- Provides better way of classification of actions which are taken by the antivirus.
- It can be used to analyse whether the antivirus can protect detect with every security level like firewall, intrusion, behaviour etc.

VI. CONCLUSION

In this paper, we systematically extract the text from images by using connected components method. Also we propose the way to analyse and classify the extracted text according to the antivirus protection level under which it is detected. This will be useful for the software testing team in an antivirus company in order to compare their own products with others so, they can have an idea about how much security they can provide and also the quality of their product.

REFERENCES

[1] Text Extraction from Hetrogenous ImagesUsngMathmetical Morphology- G. RAMA MOHAN BABU, P. SRIMAIYEE, A. SRIKRISHNA
[2] A New Approach forVideo Text Detection. In Proc. of International ConferenceOn Image Processing, -M. Cai, J. Song and M. R. Lyu. Rochester, New York,USA, pp. 117-120, 2002.

[3] Method for Extracting Product Information from TV Commercial- Kohei Arai, Herman Tolle

[4] Text detection in images using sparse representation with discriminative dictionaries- Ming Zhao ,Shutao Li , James Kwok Ming Zhao, Shutao Li, James Kwok

[5] A Robust Algorithm for Text Detection in Images- JulindaGllavata, Ralph Ewerth and Bernd Freisleben

Mr. Sachin Jadhav, Professor,Department of Computer Engineering, PimpriChinchwad College of Engineering, Pune University, Pune, Maharashtra, India.

Mr. Shrikant Ganmukhe, Bachelor of computer engineering, PimpriChinchwad College of Engineering, Pune University, Pune, Maharashtra, India.

Mr. Sanket Badwe, Bachelor of computer engineering, PimpriChinchwad College of Engineering, Pune University, Pune, Maharashtra, India.

Mr. Bhushan Bhavsar, Bachelor of computer engineering, PimpriChinchwad College of Engineering, Pune University, Pune, Maharashtra, India.